09-07-00 A

Please type a plus sign (+) inside this box → [+]

PTO/SB/05 (4/98)
Approved for use through 09/30/2000. OMB 0651-0032
Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

UTILITY PATENT APPLICATION TRANSMITTAL

(Only for new nonprovisional applications under 37 C.F.R. § 1.53(b))

Attorney Docket No.	204,797
First Inventor or Application Identifier	Jean-Francois MOYERSOEN
Title	METHOD OF OFFERING FREE PRODUCTS OR SERVICES OVER THE INTERNET
Express Mail Label No.	EJ620981207US

APPLICATION ELEMENTS

See MPEP chapter 600 concerning utility patent application contents.

ADDRESS TO: Assistant Commissioner for Patents, Box Patent Application, Washington, DC 20231

1. [X] * Fee Transmittal Form *(e.g., PTO/SB/17)* *(Submit an original and a duplicate for fee processing)*
2. [X] Specification [Total Pages 17] *(preferred arrangement set forth below)*
   - Descriptive title of the Invention
   - Cross References to Related Applications
   - Statement Regarding Fed sponsored R & D
   - Reference to Microfiche Appendix
   - Background of the Invention
   - Brief Summary of the Invention
   - Brief Description of the Drawings *(if filed)*
   - Detailed Description
   - Claim(s)
   - Abstract of the Disclosure
3. [X] Drawing(s) *(35 U.S.C. 113)* [Total Sheets 13]
4. Oath or Declaration unsigned [Total Pages 2]
   a. [XX] Newly executed (original or copy)
   b. [ ] Copy from a prior application (37 C.F.R. § 1.63(d)) *(for continuation/divisional with Box 16 completed)*
      i. [ ] DELETION OF INVENTOR(S) Signed statement attached deleting inventor(s) named in the prior application, see 37 C.F.R. §§ 1.63(d)(2) and 1.33(b).

NOTE FOR ITEMS 1 & 13: IN ORDER TO BE ENTITLED TO PAY SMALL ENTITY FEES, A SMALL ENTITY STATEMENT IS REQUIRED (37 C.F.R. § 1.27), EXCEPT IF ONE FILED IN A PRIOR APPLICATION IS RELIED UPON (37 C.F.R. § 1.28).

5. [ ] Microfiche Computer Program *(Appendix)*
6. Nucleotide and/or Amino Acid Sequence Submission *(if applicable, all necessary)*
   a. [ ] Computer Readable Copy
   b. [ ] Paper Copy (identical to computer copy)
   c. [ ] Statement verifying identity of above copies

ACCOMPANYING APPLICATION PARTS

7. [ ] Assignment Papers (cover sheet & document(s))
8. [ ] 37 C.F.R.§3.73(b) Statement *(when there is an assignee)* [X] Power of Attorney
9. [ ] English Translation Document *(if applicable)*
10. [ ] Information Disclosure Statement (IDS)/PTO-1449 [ ] Copies of IDS Citations
11. [ ] Preliminary Amendment
12. [ ] Return Receipt Postcard (MPEP 503) *(Should be specifically itemized)*
13. [ ] * Small Entity Statement(s) *(PTO/SB/09-12)* [X] Statement filed in prior application, Status still proper and desired
14. [ ] Certified Copy of Priority Document(s) *(if foreign priority is claimed)*
15. [ ] Other: .....

16. **If a CONTINUING APPLICATION,** *check appropriate box, and supply the requisite information below and in a preliminary amendment:*

[ ] Continuation [ ] Divisional [ ] Continuation-in-part (CIP) of prior application No: _____/_____

Prior application information: Examiner _____ Group / Art Unit: _____

For CONTINUATION or DIVISIONAL APPS only: The entire disclosure of the prior application, from which an oath or declaration is supplied under Box 4b, is considered a part of the disclosure of the accompanying continuation or divisional application and is hereby incorporated by reference. The incorporation can only be relied upon when a portion has been inadvertently omitted from the submitted application parts.

17. CORRESPONDENCE ADDRESS

[ ] *Customer Number or Bar Code Label* *(Insert Customer No. or Attach bar code label here)* or [X] *Correspondence address below*

Name	ABELMAN FRAYNE & SCHWAB Attorneys at Law				
Address	150 East 42nd Street New York, NY 10017				
City		State		Zip Code	
Country	U.S.A.	Telephone	(212) 949-9022	Fax	(212) 949-9190

Name (Print/Type)	Michael J. Markowitz	Registration No. (Attorney/Agent)	30,659
Signature		Date	Sep. 6, 2000

Burden Hour Statement: This form is estimated to take 0.2 hours to complete. Time will vary depending upon the needs of the individual case. Any comments on the amount of time you are required to complete this form should be sent to the Chief Information Officer, Patent and Trademark Office, Washington, DC 20231. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Assistant Commissioner for Patents, Box Patent Application, Washington, DC 20231.

PTO/SB/17 (6/99)
Approved for use through 09/30/2000 OMB 0651-0032
Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

FEE TRANSMITTAL for FY 1999

Patent fees are subject to annual revision.
Small Entity payments must be supported by a small entity statement, otherwise large entity fees must be paid. See Forms PTO/SB/09-12. See 37 C.F.R. §§ 1.27 and 1.28.

TOTAL AMOUNT OF PAYMENT ($) 345.00

Complete if Known	
Application Number	
Filing Date	September 6, 2000
First Named Inventor	Jean-Francois MOYERSOEN
Examiner Name	
Group / Art Unit	
Attorney Docket No.	204,797

METHOD OF PAYMENT (check one)

1. [X] The Commissioner is hereby authorized to charge indicated fees and credit any over payments to

Deposit Account Number: 01-0035

Deposit Account Name: ABELMAN, FRAYNE & SCHWAB

[X] Charge Any Additional Fee Required Under 37 CFR §§ 1.16 and 1.17

2. [X] Payment Enclosed:
[X] Check [ ] Money Order [ ] Other

FEE CALCULATION

1. BASIC FILING FEE

Large Entity Fee Code	Large Entity Fee ($)	Small Entity Fee Code	Small Entity Fee ($)	Fee Description	Fee Paid
101	760	201	380	Utility filing fee	345.00
106	310	206	155	Design filing fee	
107	480	207	240	Plant filing fee	
108	760	208	380	Reissue filing fee	
114	150	214	75	Provisional filing fee	

SUBTOTAL (1) ($) 345.00

2. EXTRA CLAIM FEES

		Extra Claims		Fee from below		Fee Paid
Total Claims		-20** =		X	=	
Independent Claims		- 3** =		X	=	
Multiple Dependent					=	

**or number previously paid, if greater; For Reissues, see below

Large Entity Fee Code	Large Entity Fee ($)	Small Entity Fee Code	Small Entity Fee ($)	Fee Description
103	18	203	9	Claims in excess of 20
102	78	202	39	Independent claims in excess of 3
104	260	204	130	Multiple dependent claim, if not paid
109	78	209	39	** Reissue independent claims over original patent
110	18	210	9	** Reissue claims in excess of 20 and over original patent

SUBTOTAL (2) ($)

FEE CALCULATION (continued)

3. ADDITIONAL FEES

Large Entity Fee Code	Large Entity Fee ($)	Small Entity Fee Code	Small Entity Fee ($)	Fee Description	Fee Paid
105	130	205	65	Surcharge - late filing fee or oath	
127	50	227	25	Surcharge - late provisional filing fee or cover sheet.	
139	130	139	130	Non-English specification	
147	2,520	147	2,520	For filing a request for reexamination	
112	920*	112	920*	Requesting publication of SIR prior to Examiner action	
113	1,840*	113	1,840*	Requesting publication of SIR after Examiner action	
115	110	215	55	Extension for reply within first month	
116	380	216	190	Extension for reply within second month	
117	870	217	435	Extension for reply within third month	
118	1,360	218	680	Extension for reply within fourth month	
128	1,850	228	925	Extension for reply within fifth month	
119	300	219	150	Notice of Appeal	
120	300	220	150	Filing a brief in support of an appeal	
121	260	221	130	Request for oral hearing	
138	1,510	138	1,510	Petition to institute a public use proceeding	
140	110	240	55	Petition to revive - unavoidable	
141	1,210	241	605	Petition to revive - unintentional	
142	1,210	242	605	Utility issue fee (or reissue)	
143	430	243	215	Design issue fee	
144	580	244	290	Plant issue fee	
122	130	122	130	Petitions to the Commissioner	
123	50	123	50	Petitions related to provisional applications	
126	240	126	240	Submission of Information Disclosure Stmt	
581	40	581	40	Recording each patent assignment per property (times number of properties)	
146	760	246	380	Filing a submission after final rejection (37 CFR § 1.129(a))	
149	760	249	380	For each additional invention to be examined (37 CFR § 1.129(b))	

Other fee (specify) _____

Other fee (specify) _____

*Reduced by Basic Filing Fee Paid

SUBTOTAL (3) ($)

SUBMITTED BY

				Complete (if applicable)	
Name (Print/Type)	Michael I. Markowitz	Registration No. (Attorney/Agent)	30,659	Telephone	(212) 949-9022
Signature				Date	Sep. 6, 2000

Burden Hour Statement: This form is estimated to take 0.2 hours to complete. Time will vary depending upon the needs of the individual case. Any comments on the amount of time you are required to complete this form should be sent to the Chief Information Officer, Patent and Trademark Office, Washington, DC 20231. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Assistant Commissioner for Patents, Washington, DC 20231.

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICANT: MOYERSOEN

SERIAL NO.: N/A EXAMINER:

FILED: May 16, 2000 GROUP NO.:

FOR (TITLE): METHOD OF OFFERING FREE PRODUCTS OR SERVICES OVER THE INTERNET

VERIFIED STATEMENT AS SMALL ENTITY

Hon. Commissioner of Patents and Trademarks
Washington, D.C. 20231

SIR:

The undersigned declare(s):

Exclusive rights in the above-identified inventior named below, and "small entity" fees are appropriate. Qualificati following:

STATEMENT OF FILING BY EXPRESS MAIL (37 CFR § 1.10)
This correspondence is being deposited with the United States Postal Service on September 06, 2000 in an envelope as "Express Mail Post Office to Addressee" Mailing Label Number EJ620981207US addressed to Director of the U.S. Patent and Trademark Office, Washington, D.C. 20231.

[X] **INDEPENDENT INVENTOR**

An independent inventor is any inventor who:

1) has not assigned, granted, conveyed, or licensed, and

2) is under no obligation under contract or law to assign, grant, convey, or license any rights in the invention to any person who could not likewise be classified as an independent inventor if that person had made the invention, or to any concern which would not qualify as a small business concern or a non-profit organization as defined in Rule 1.9.

[ ] **SMALL BUSINESS CONCERN**

A small business concern is defined as a business concern:

1) whose number of employees, including those of its affiliates, does not exceed 500 persons, and

2) which has not assigned, granted, conveyed, or licensed, and is under no obligation under contract or law to assign, grant, convey or license, any rights in the invention to any person who could not be classified as an independent inventor if that person had made the invention, or to any concern which would not qualify as a small business concern or a nonprofit organization as defined in Rule 1.9. Concerns are affiliates of each other when, either directly or indirectly, one concern controls or has the power to control the other, or a third party controls or has the power to control both. The number of employees of the business concern is the average over the fiscal year of the persons employed during each of the pay periods of the fiscal year. Employees are those persons employed on a full-time, part-time or temporary basis during the previous fiscal year of the concern.

☐ NONPROFIT ORGANIZATION (Check additional applicable box)

A nonprofit organization is defined as:

☐ 1) a university or other institution of higher education located in any country; or

☐ 2) a organization of the type described in section 501(c)(3) of the Internal Revenue Code of 1954 (26 U.S.C. 501(c)(3)) and exempt from taxation under Section 501(a) of the Internal Revenue Code (26 U.S.C. 501(a)); or

☐ 3) any nonprofit scientific or educational organization qualified under a nonprofit organization statute of a state of the United States (35 U.S.C. 201(i)); or

☐ 4) any nonprofit organization located in a foreign country which would qualify as a nonprofit organization under paragraphs (e)(2) or (3) of Rule 1.9 if it were located in the United States.

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

Jean-Francois MOYERSOEN
(Print or type name of small entity)

* _____
(Signature and title of person authorized to act on behalf of small entity)

27 May 2000
(Date)

31, Avenue Princesse Grace
(Address)
MC- 98000 MONACO

* If "small entity" is more than one inventor or a combination of "small entities", add name(s), address(es) and signature(s) of other parties below. Please include date.

Full Name of Second Joint Inventor, If Any	Inventor's Signature	Date
Residence	Citizenship	
Post Office Address		
Full Name of Third Joint Inventor, If Any	Inventor's Signature	Date
Residence	Citizenship	
Post Office Address		
Full Name of Fourth Joint Inventor, If Any	Inventor's Signature	Date
Residence	Citizenship	
Post Office Address		

METHOD OF OFFERING FREE PRODUCTS OR SERVICES OVER THE INTERNET

CROSS-REFERENCE TO RELATED APPLICATIONS

This application claims the benefit of the filing date of U.S. Provisional Application No. 60/204,801, filed on May 16, 2000.

BACKGROUND OF THE INVENTION

FIELD OF THE INVENTION

This invention relates to commerce conducted over offer of products or services for sale over the Internet.

DESCRIPTION OF THE RELATED ART

The purchase and sale of goods and services over growing at a rapid rate. Although there are many advant such electronic commerce transactions, such as the low co the ease of employment of the Internet, the virtually unlimited range of products and services offered, the security of payments, and the anonymity of transactions, many consumers are still hesitant to purchase over the Internet due to privacy or security concerns, the novelty of the methods involved, or for many other unknown or not easily articulated reasons.

The subject invention offers an additional incentive to hesitant consumers to purchase over the Internet, besides the other advantages previously cited. The invention provides this incentive by allowing a consumer of goods or services to acquire those goods or services at

no cost according to a random process, and the consumer can be notified, prior to an order, of the probability that he or she will receive the goods or services desired to be purchased at no cost.

Thus, for example, a consumer may be informed that if he wishes to purchase a particular book, every tenth book ordered will be delivered at no charge. Alternatively, the consumer may be informed that the number of books ordered before a free one is offered will be a random number. In addition, a consumer may be notified of the number of purchase orders placed for a particular good or service.

SUMMARY OF THE INVENTION

The invention comprises a method of offering free product(s) and/or service(s) over the Internet embodied by, for example, a computer software program for E-Commerce applications (hereinafter sometimes referred to as "**LUCK(Y)CYCLE**"). The program is an enhancement to existing merchant web-sites which would enable the merchant to offer free product(s) and/or service(s) to customers in accordance with pre-set parameters chosen by the merchant as part of his marketing strategy.

The software program comprises four user-definable algorithms allowing the merchant user to predict the probability of a free product and/or service being offered to the customer.

BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 is a flow chart of an Internet purchase transaction without the use of the subject invention.

Fig. 2 is a flow chart of an internet purchase transaction utilizing the subject invention.

Figs. 3 and 3A are source code for the default **LUCK(Y)CYCLE** management screen available for each individual and/or (a) group(s) of product(s) and/or service(s) in an online store's catalog.

Figs. 4, 4A, and 4B are source code for result screens for **LUCK(Y)CYCLE** showing the winning product(s) and/or service(s), depending on the merchant's choice of one of the four user-definable algorithms.

Figs. 5, 5A, and 5B are source code for the four **LUCK(Y)CYCLE** user-definable algorithms.

Fig. 6 shows a **LUCK(Y)CYCLE** management screen where the merchant has selected the regular cycle algorithm for a particular individual and/or (a) group(s) of product(s) and/or service(s) in the on-line store's catalog.

Fig. 7 shows a **LUCK(Y)CYCLE** result screen showing the individual and/or (a) group(s) of product(s) and/or service(s) offered for free corresponding to the merchant's choice of the regular cycle algorithm for a particular individual and/or (a) group(s) of product(s) and/or service(s) as shown in Fig. 6.

Fig. 8 shows a **LUCK(Y)CYCLE** management screen where the merchant has selected the constant probability algorithm for a particular individual and/or (a) group(s) of product(s) and/or service(s) in the on-line store's catalog.

Fig. 9 shows a **LUCK(Y)CYCLE** result screen showing the individual and/or (a) group(s) of product(s) and/or service(s) offered for free corresponding to the merchant's

choice of the constant probability algorithm for a particular individual and/or (a) group(s) of product(s) and/or service(s) as shown in Fig. 8.

Fig. 10 shows a **LUCK(Y)CYCLE** management screen where the merchant has selected the pre-defined list algorithm for a particular individual and/or (a) group(s) of product(s) and/or service(s) in the on-line store's catalog.

Fig. 11 shows a **LUCK(Y)CYCLE** result screen showing the individual and/or (a) group(s) of product(s) and/or service(s) offered for free corresponding to the merchant's choice of the pre-defined list algorithm for a particular individual and/or (a) group(s) of product(s) and/or service(s) as shown in Fig. 10.

Fig. 12 shows a **LUCK(Y)CYCLE** management screen where the merchant has selected the dynamic probability algorithm for a particular individual and/or (a) group(s) of product(s) and/or service(s) in the on-line store's catalog.

Fig. 13 shows a **LUCK(Y)CYCLE** result screen showing the individual and/or (a) group(s) of product(s) and/or service(s) offered for free corresponding to the merchant's choice of the dynamic probability algorithm for a particular individual and/or (a) group(s) of product(s) and/or service(s) as shown in Fig. 12.

DESCRIPTION OF THE PREFERRED EMBODIMENT(S)

E-Commerce and On-line Shopping

E-commerce sites (such as Amazon.com for example) offer customers a large catalog of individual and/or (a) group(s) of products and/or services. As shown in Fig. 1, on

arriving at the on-line shop 1 on the internet using a web browser 2, the customer may browse 4 through the range of products and/or services available for purchase in the on-line catalog 6, progressively select product(s) and/or service(s) for purchase 8, and add them to a virtual "shopping basket" 10 prior to payment 18 (via credit card 16) and exit 20 from the "store" 1.

As shown in Fig. 2, with the addition of the **LUCK(Y)CYCLE** program 22 to an on-line store, the customer could be entitled to benefit from free offers, free products and/or services, etc. 24 in accordance with a strategy pre-defined by the merchant and regulated and managed by the Lucky Cycle program.

The **LUCK(Y)CYCLE** program will enable the merchant to attribute to each individual and/or (a) group(s) of product(s) and/or service(s) in his catalog individualized parameters which will decide the probability of a free gift of that individual and/or (a) group(s) of product(s) and/or service(s) 26 being offered to the customer.

How it Works

General Principle

Each individual and/or group of product(s) and/or service(s) can be characterized by its own algorithmic cycle which will define the statistical probability of it being offered free to the customer. This cycle is pre-defined by the merchant and represents an estimate of the number of items which should be sold in order that one item or group of items may be offered free.

Procedures activated at the moment of purchase of any item or group of items will enable the customer to see whether he receives it or them for free or whether he must pay for it or them.

In this application, the letter "n" will be used to represent the cycle selected by the merchant. The proposed algorithms will be based, amongst other things, on the number of catalog items of any given type ordered by the total number of customers visiting the site since its opening. Each catalog item ordered will thus have its own index, which is "p", and the cycle selected by the merchant is specific to each catalog item.

Example

Assume that an E-commerce site is offering two articles, A and B.

Article A has a cycle where n=10, which means that the probability of it being offered for free is 1/10 or 10%.

Article B has a cycle independent of article A, and which may be different from n=10.

The first article A has an index of p=1.

The second article A has an index of p=2.

The first article B has an index of p=1.

And so on...

Explanation of the Different Algorithms

The regular cycle

In this algorithm, after (n-1) articles have been sold, the nth article is offered for free. The probability is therefore a direct function of "p".

Mathematically, it could be stated that the article is offered for free when (p modulus n) = 0.

This mathematical statement could be extended to a more general equation: (p modulus n) = whole number constant between 0 and (n-1). If we call the whole number constant c, this more general equation would describe the result that after c-1 articles have been sold, the cth article is offered for free for the first cycle, after n+c-1 articles have been sold, the (n+c)th article is offered for free for the second cycle, and so on.

The source code of the regular cycle principle is shown in Fig. 5 underneath the highlighted regular cycle algorithm title.

A screenshot in Fig. 6 shows how the merchant selects this algorithm. The screenshot in Fig. 7 shows the resulting individual and/or group of products and/or services offered for free when n has been set to 10 and p has been set to 50.

Example

The cycle where n=10 would mean that the 10th, the 20th, the 30th...etc...article ordered would be offered free to the customer.

The constant probability

This algorithm is characterized by an identical probability for all values of "p". For all orders placed for the article, each customer will have a 1/n probability of a free gift.

Mathematically, this cycle is characterized by the generation of a random number between 0 and (n-1). If this number equals 0 (or any other constant between 0 and (n-1)), then the article is offered for free.

The source code of the constant probability algorithm is shown in Fig. 5 underneath the highlighted constant probability algorithm title.

A screenshot in Fig. 8 shows how the merchant selects this algorithm. Fig. 9 shows the resulting individual and/or group of products and/or services offered for free when n has been set to 10 and p has been set to 50.

Example

The cycle where n=10 would give all customers for this article a 1 in 10 chance of winning it for free.

The pre-defined list

This algorithm comprises determining at the opening of the site a series of whole numbers included between 1 and v which will determine future winners.

If the index "p" for any particular order corresponds to a number contained within this list, then the article is offered for free.

This list should therefore contain v/n numbers in order to respect the n cycle. When the v articles have been ordered, a new series of numbers must be created between v+1 and 2v.

This series of numbers may be created manually by the site administrator, or at random by a number generator.

Mathematically, v/n distinct numbers are generated with values between 1 and v. If "p" is included in this series, then the article is offered for free.

One particular case in this cycle is where v=n. In this case, the list is comprised of a single element. This "list" is recreated whenever "p" reaches a multiple of n and includes a number to be found between p and (p+n).

The source code of the pre-defined list algorithm is shown in Figs. 5 and 5A underneath the highlighted pre-defined list algorithm title.

A screenshot in Fig. 10 shows how the merchant selects this algorithm. Fig. 11 shows the resulting individual and/or group of products and/or services offered for free when n has been set to 10 and p has been set to 50.

The dynamic probability

This algorithm calculates the probability of obtaining an article for free according to the difference between p and the next article to be found in a pre-defined reference list.

For example, assume a pre-defined reference list of a regular series such as: 10, 20, 30, 40, ... corresponding to a regular cycle where n=10.

At the opening of the site, the next article in the reference list is thus 10. The first article ordered will have a probability of 1 in 10. The second article ordered will have a probability of 1 in 9. The third article ordered will have a probability of 1 in 8. If we assume that this third article is offered for free, then the next available number in the reference list becomes 20. Thus, the fourth article ordered will have a probability of 1 in 17.

If p' is the next number in the reference list, then the probability is expressed as $1/(p'-p+1)$. This algorithm can be generalized by taking any reference list, as long as it always respects the probability of 1/n. The function of probability $1/(p'-p+1)$ can itself be replaced by any other function of p and p'.

The source code of the dynamic probability algorithm is shown in Fig. 5A underneath the highlighted dynamic probability algorithm title.

A screenshot in Fig. 12 shows how the merchant selects this algorithm. Fig. 13 shows the resulting individual and/or group of products and/or services offered for free when n has been set to 10 and p has been set to 50.

It should be understood that where any of the constant probability, pre-defined list, and dynamic probability algorithms require the generation of a random number, that random number need not be an integer within the desired range of values, but may be a rational fraction as well. The fraction could then be rounded to an integer for further use in the algorithm. The use of such fractional values would have the effect of increasing the possible number of random values generated, but it should not have any effect on the probability of any integer being chosen.

What I claim is:

1. A method of selling and purchasing at least one object of purchase over a computer network, said method comprising the following steps:

a. utilizing a software program on a computer of a purchaser to search for and find a site on said computer network offering said at least one object of purchase;

b. browsing through an on-line catalog at said site to find said at least one object of purchase, each of said at least one object of purchase having attributed to it a probability of obtaining said each of said at least one object of purchase at no cost to said purchaser;

c. selecting said at least one object of purchase for purchase;

d. confirming an order for said at least one object of purchase;

e. determining whether payment must be made for said each of said at least one object of purchase; and

f. paying for only those objects of purchase from said at least one object of purchase for which payment was determined to be required in step e.

2. A method of selling and purchasing at least one object of purchase over a computer network as claimed in claim 1, wherein said step of determining whether payment must be made for said each of said at least one object of purchase comprises the following steps:

a. determining the number of said each of said at least one object of purchase ordered since an event selected from the group of events consisting of:

(i) said each of said at least one object of purchase was delivered at no cost to a prior purchaser; and

(ii) said each of said at least one object of purchase was first offered for sale;

b. offering said each of said at least one object of purchase to said purchaser at no cost if said number is equal to a predetermined value; and

c. requiring payment for said each of said at least one object of purchase if said number is not equal to said predetermined value.

3. A method of selling and purchasing at least one object of purchase over a computer network as claimed in claim 1, wherein said step of determining whether payment must be made for said each of said at least one object of purchase comprises the following steps:

a. generating a random number between a first predetermined value and a second predetermined value for said each of said at least one object of purchase;

b. offering said each of said at least one object of purchase to said purchaser at no cost if said random number is equal to a third predetermined value; and

c. requiring payment for said each of said at least one object of purchase if said random number is not equal to said third predetermined value.

4. A method of selling and purchasing at least one object of purchase over a computer network as claimed in claim 1, wherein said step of determining whether payment must be made for said each of said at least one object of purchase comprises the following steps:

a. determining a series of whole numbers for said each of said at least one object of purchase after a first event selected from the group of events consisting of:

(i) said site is ready to sell for the first time said each of said at least one object of purchase; and

(ii) a first predetermined number of said each of said at least one object of purchase have been ordered after a second event selected from the group of events consisting of:

(a) said site is ready to sell for the first time said each of said at least one object of purchase; and

(b) determining a series of whole numbers as specified in this step a., said series of whole numbers being between one and said first predetermined number,

said series of whole numbers having as many numbers as the integral part of the quotient of said first predetermined number divided by a second predetermined number;

b. offering said each of said at least one object of purchase to said purchaser at no cost if the number of said each of said at least one object of purchase ordered since step a. was last performed is equal to one of said series of whole numbers; and

c. requiring payment for said each of said at least one object of purchase if said number of said each of said at least one object of purchase ordered since step a. was last performed is not equal to any of said series of whole numbers.

5. A method of selling and purchasing at least one object of purchase over a computer network as claimed in claim 1, wherein said step of determining whether payment must be made for said each of said at least one object of purchase comprises the following steps:

a. determining a series of whole numbers for said each of said at least one object of purchase after a first event selected from the group of events consisting of:

(i) said site is ready to sell for the first time said each of said at least one object of purchase; and

(ii) a first predetermined number of said each of said at least one object of purchase have been ordered after a second event selected from the group of events consisting of:

(a) said site is ready to sell for the first time said each of said at least one object of purchase; and

(b) determining a series of whole numbers as specified in this step a., said series of whole numbers being between one and said first predetermined number,

said series of whole numbers having as many numbers as the integral part of the quotient of said first predetermined number divided by a second predetermined number;

b. generating a random number between zero and a first number of said each of said at least one object of purchase that must be ordered in the future for a second number of said each of said at least one object of purchase ordered since step a. was last performed to equal the least of said series of whole numbers which is greater than a fourth number of said each of said at least one object of purchase actually ordered since step a. was last performed, said generation of said random number being performed for said each of said at least one object of purchase;

c. offering said each of said at least one object of purchase to said purchaser at no cost if said random number is equal to zero; and

d. requiring payment for said each of said at least one object of purchase if said random number is not equal to zero.

6. A system for selling at least one object of purchase over a computer network, said system comprising:

a. an on-line catalog;

b. a virtual shopping basket; and

c. software for determining whether a particular one of said at least one object of purchase shall be offered free to a purchaser.

7. A system for selling at least one object of purchase over a computer network as claimed in claim 6, wherein said software comprises:

a. a program to produce a user interface allowing a merchant to select one of at least one algorithm for determining whether a particular one of said at least one object of purchase shall be offered free to said purchaser, said selection being with regard to one object of purchase in said on-line catalog;

b. said at least one algorithm; and

c. a program to produce a user interface displaying to a merchant the particular one(s) of said one object of purchase in said on-line catalog offered free to purchasers.

8. A system for selling at least one object of purchase over a computer network as claimed in claim 7, wherein said at least one algorithm comprises a regular cycle algorithm, a constant probability algorithm, a pre-defined list algorithm, and a dynamic probability algorithm.

9. A system for selling at least one object of purchase over a computer network as claimed in claim 6, wherein said on-line catalog notifies a potential purchaser of the probability of obtaining at least one item therein at no cost.

ABSTRACT OF THE DISCLOSURE

A system and method for offering free goods or services over computer networks allows a merchant to choose a probability-based method of selecting how the free items are determined. The probability of obtaining a free item for any item desired can also be displayed to a potential purchaser.

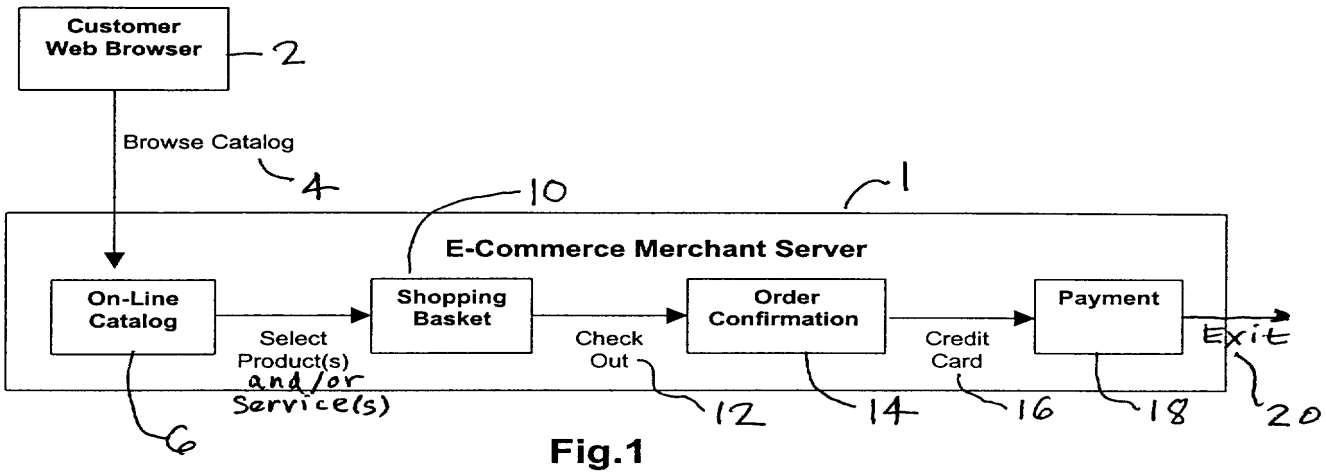
Customer Web Browser
2
Browse Catalog
4
10
1
E-Commerce Merchant Server
On-Line Catalog
Shopping Basket
Order Confirmation
Payment
Exit
Select Product(s) and/or Service(s)
Check Out
Credit Card
6
12
14
16
18
20

Fig.1

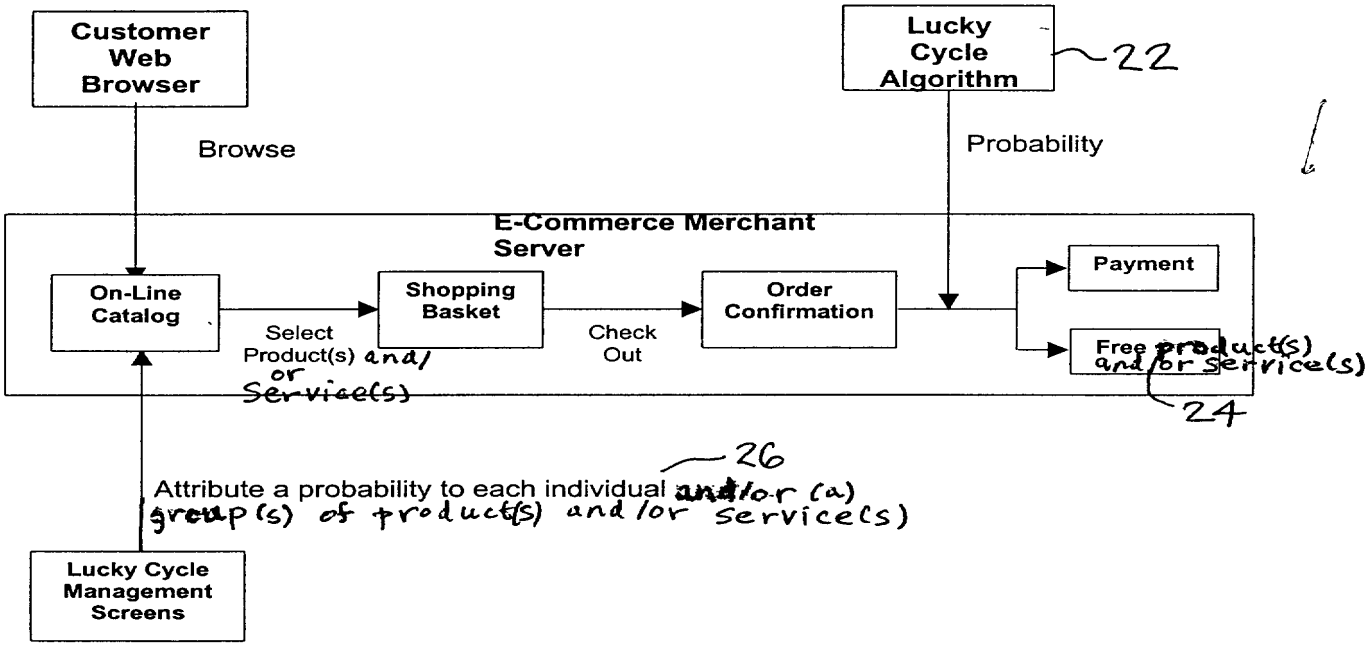
Customer Web Browser
Lucky Cycle Algorithm
22
Browse
Probability
E-Commerce Merchant Server
On-Line Catalog
Shopping Basket
Order Confirmation
Payment
Free product(s) and/or service(s)
Select Product(s) and/or Service(s)
Check Out
24
26
Attribute a probability to each individual and/or (a) group(s) of product(s) and/or service(s)
Lucky Cycle Management Screens

Fig.2

```
<!--
      Lucky Cycle
      March 2000
      JF MOYERSOEN

      Data Entry Form
-->


<html>
<head>
      <title>Lucky Cycle</title>
</head>

<body>
<center>
<font face=verdana size=3><b>Lucky Cycle</b></font><b>
<form action="result.asp" method=post>
<!-- Display of an Error Message, followed by the initialisation of
this Error Message -->
<br><font face=verdana size=2>
Concept invented and registered by Jean-François Moyersoen
<p> </p>
    <table bgcolor="#E0E0E0" border=1 cellpadding=20 width="461">
      <tr><td>
            <font face=verdana size=2 color=red><i><%=
Session("error_message") %>
            <% Session("error_message") = "" %>
              <table width="419">
            <tr>
                  <td colspan=2><font face=verdana size=2><b>Selected
Algorithm :</b></font></td>
            </tr>
            <tr>
                  <td align=right width="162">
              <input type=radio name="algorithm" value="1" <% If
Session("algorithm") = "1" then response.write(" checked ") %>></td>
                  <td width="245"><font face=verdana size=2>The
regular cycle</font></td>
            </tr>
            <tr>
                  <td align=right width="162">
                <input type=radio name="algorithm" value="2" <% If
Session("algorithm") = "2" then response.write(" checked ") %>></td>
                  <td width="245"><font face=verdana size=2>The
constant probability</font></td>
            </tr>
            <tr>
                  <td align=right width="162">
                <input type=radio name="algorithm" value="3" <% If
Session("algorithm") = "3" then response.write(" checked ") %>></td>
                  <td width="245"><font face=verdana size=2>The
pre-defined list</font></td>
            </tr>
            <tr>
                  <td align=right width="162">
                <input type=radio name="algorithm" value="4" <% If
Session("algorithm") = "4" then response.write(" checked ") %>></td>
                  <td width="245"><font face=verdana size=2>The
dynamic probability</font></td>
```

Fig. 3

```
            </tr>
            <tr>
                  <td colspan=2> </td>
            </tr>
            <tr>
                  <td colspan=2><font face=verdana
size=2><b>Parameters :</b></font></td>
            </tr>
            <tr>
                    <td width="162"><font face=verdana
size=2>Cycle</font></td>
                    <td width="245"><font face=verdana size=2>n=
                <input type=text name="n" maxlength=3 size=3
value="<%= Session("n") %>"></font></td>
            </tr>
            <tr>
                    <td width="162"><font face=verdana size=2>Number
of purchases </font></td>
                    <td width="245"><font face=verdana size=2>p=
                <input maxlength=4 type=text name="pmax" size=3
value="<%= Session("pmax") %>"></font></td>
            </tr>
            <tr>
                  <td colspan=2> </td>
            </tr>
            <tr>
                  <td colspan=2 align=center><input type=submit
value="Simulation"></td>
            </tr>
            </table>
</i></font></td></tr>
</table>
</font></form>
</b></center></body>
</html>
```

Fig. 3A

```
<!-- #include file="algorithm.inc" -->
<%
      '## Input of the form data if the form is not empty
      '## If this page is referred to by a page other than
default.asp, this form does not exist
      '## and the instruction bloc will not be executed
      If Request.form("n") <> "" or Request.Form("pmax") <> "" or
Request.Form("algorithm") <> "" Then
            Session("n") = Trim(Request.Form("n"))
            Session("pmax") = Trim(Request.Form("pmax"))
            Session("algorithm") = Trim(Request.Form("algorithm"))
      End If

      '## Verification of the selected algorithm
      If Session("algorithm") <> "1" and Session("algorithm") <> "2"
and Session("algorithm") <> "3" _
            and Session("algorithm") <> "4" Then Return_Error ("The
algorithm is not correct")

      '## Verification if the value N has been entered
      If Session("n") = "" then Return_Error("N is empty")
      If not Isnumeric(Session("n")) then Return_Error("N is not a
number")
      If Cstr(CLng(Session("n"))) <> Session("n") then
Return_Error("N is not a whole number")
      If CLng(Session("n")) <= 0 Then Return_Error("N must be a
positive number")

      '## Verification of the entered Pmax value
      If Session("pmax") = "" then Return_Error("Pmax is empty")
      If not Isnumeric(Session("pmax")) then Return_Error("Pmax is
not a number")
      If Cstr(CLng(Session("pmax"))) <> Session("pmax") then
Return_Error("Pmax is not a whole number")
      If CLng(Session("pmax")) <= 0 Then Return_Error("Pmax must be a
positive number")

      '## Initialisation of the variables
      nb_articles_won = 0
      Randomize()


      '## Return function to the previous page if an error occurs
      '## the Error Message is stored in the Session("Error_Message")
      Sub Return_Error(p_message)
            Session("Error_Message") = p_message
            response.buffer = true
            response.clear
            response.redirect("default.asp")
            response.end
      End Sub
```

Fig. 4

```
'## Display of the results table
      Sub Table()

            '## Selected algorithm by the
Session("algorithm")variable
            Select Case Session("algorithm")

            '## For each algorithm, the index of the ordered article
p varies between 1 and Pmax
            '## For each value p, a function containing the Lucky
Cycle algorithm is called
            '## The parameters to be passed to these different
functions are the cycle n stored in the Session("n") and p
            '## The result is False if the ordered product is not
given for free and True if the product is a free gift

            '## The cell function displays a cell of the table
            '## The parameters to be passed are the index p to be
displayed inside the cell and
            '## the return value of the algorithm that will define
the background color of the cell
            Case "1" :        For p = 1 to Session("pmax")
                                    Cell p,
algorithm_1(Session("n"), p)
                              Next
            Case "2" :        For p = 1 to Session("pmax")
                                    Cell p,
algorithm_2(Session("n"), p)
                              Next
            Case "3" :        For p = 1 to Session("pmax")
                                    Cell p,
algorithm_3(Session("n"), p)
                              Next
            Case "4" :        For p = 1 to Session("pmax")
                                    Cell p,
algorithm_4(Session("n"), p)
                              Next

            End Select

      End Sub

      '## Display of the table cell with a result
      Sub Cell(index_p, reponse_algorithm)

            '## If the cell is the first in a serie of 20, the
following end of line/begin of line tags will be inserted
            if index_p mod 20 = 1 then
                  response.write("</tr><tr>")
            end if

            '## If the index corresponds to a free product, the
background and text color will be defined
            if reponse_algorithm = true then
                  bg_color = "red"
                  text_color = "white"
                  '## The number of articles won is incremented
                  nb_articles_won = nb_articles_won + 1
            else
            '## If the product is not offered for free, other colors
will be used for the display
```

Fig. 4A

```
                bg_color = "white"
                text_color = "black"
            end if

            '## Display of a cell
            response.write("<td align=center bgcolor='" & bg_color &
"'>" & _
                        "<font color='" & text_color & "'
face=verdana size=2>" & index_p & "</td>")

      End Sub

%>
<html>
<head>
      <title>Lucky Cycle</title>
</head>

<body>
<table cellspacing=1 cellpadding=3>
<tr>
      <td colspan=20><font face=verdana size=2><b>Result
Table</b></font></td>
<% Call Table %>
</tr>
<tr>
      <td colspan=20><br><font face=verdana size=2><b><%=
nb_articles_won %> articles on <%= Session("pmax") %> have been won

      <%    '## If the number of articles is different from zero
            If nb_articles_won <> 0 Then %>
            (1 on <%= FormatNumber(Session("pmax")/nb_articles_won,3)
%>)
      <%    End If %>
      <br></b> Theoretical Cycle = <%= Session("n") %>
      <br><br>
      <form action=result.asp method=post>
      <input type=button value="Back"
onclick="document.location.href='default.asp'">
      <input type=submit value="New Simulation">
      </form>
      </font></td>
</tr>
</table>

</body>
</html>
```

Fig. 4B

```
<%

Dim p_won    '## Variable storing the index of the next article that
will be offered free
                              '## (or that will be used as a reference for
the dynamic probability algorithm)

'## All the procedures use the parameters cycle n (cycle_n) and the
index p (index_p)
'## The result of each procedure is a boolean (True if the article is
given free or False in the other situation)

'## The regular cycle
'## is based on a fixed cycle : after (n-1) articles have been sold,
the nth article is offered free
'## Mathematically, it could be stated that the article is offered
free when
'## index_p Mod cycle_n = constant number between 0 and (n-1)
'## For example : if index_p Mod cycle_n = 0

Function Algorithm_1(cycle_n, index_p)
      If index_p Mod cycle_n = 0 Then
            Algorithm_1 = True
      Else
            Algorithm_1 = False
      End If
End Function

'## The constant probability
'## The cycle is based on a constant probability of 1/n
'## Mathematically, this cycle is characterized by the generation of
a random number between 0 and (n-1)
'## If this number equals any constant between 0 and (n-1), then the
article is offered free
'## For example, if the number is equal to 0

Function Algorithm_2(cycle_n, index_p)
      nb_random = Int(cycle_n * Rnd)
      If nb_random = 0 Then
            Algorithm_2 = True
      Else
            Algorithm_2 = False
      End If
End Function
```

Fig. 5

'## The pre-defined list

```
'## This cycle is characterized by the creation of a predefined list
with all the indexes p that will be future winners
'## This list will be created on regular intervals, depending on the
number of elements defined in the list
'## This list must itself respect the cycle n and as a result the
probability 1/n.
'## The algorithm underneath represents a special case in which the
list contains only one element
'## and is thus rebuild every time n articles have been ordered
'## In this situation, this list is created by randomly assigning a
number between index_p and index_p + cycle_n

Function Algorithm_3(cycle_n, index_p)
      '## Creation of the list if the article of the index p begins
with a serie of n orders
      '## this means if the index_p mod cycle_n = 1
      '## Special case : if the cycle_n = 1 then no matter what the
value is of p,
      '## a list will be recreated (the article is the first of a
serie of 1 order), when p mod 1 <> 1
      If index_p mod cycle_n = 1 or cycle_n=1 Then
            p_won = index_p + Int(cycle_n * Rnd)
      End If
      '## If the index p is found in the list p_won containing a
single element, it will be offered free
      If index_p = p_won Then
            Algorithm_3 = True
      Else
            Algorithm_3 = False
      End If
End Function
```

'## The dynamic probability

```
'## This cycle calculates the probability of an order with index p in
function of a winning reference order,
'## that in this case would correspond to a regular cycle (see the
first algorithm)
'## The probability is calculated in function of the index_p and the
winning reference order
'## In the function underneath, we take as a reference list (n, 2*n,
3*n, 4*n, ...)
'## This list can contain any value as long as it respects itself the
cycle n and the probability 1/n

Function Algorithm_4(cycle_n, index_p)
      '## Initialisation during the first passage of p_won = cycle_n
      If index_p = 1 Then
            p_won = cycle_n
      End If
      '## Calculation of the inverse of the probability
      '## In this case, we take (p_won - index_p + 1)
      Inv_probability = (p_won - index_p + 1)
      '## Generation of a random number between 0 and
(inv_probability - 1)
      nb_random = Int(Inv_probability * Rnd)
      '## If the number is equal to 0, the product is offered free
      If nb_random = 0 Then
            Algorithm_4 = True
```

Fig. 5A

```
            '## The winning reference order takes the following index
of the reference list
            p_won = CInt(p_won) + CInt(cycle_n)
      Else
            Algorithm_4 = False
      End If
End Function
%>
```

Fig. 5B

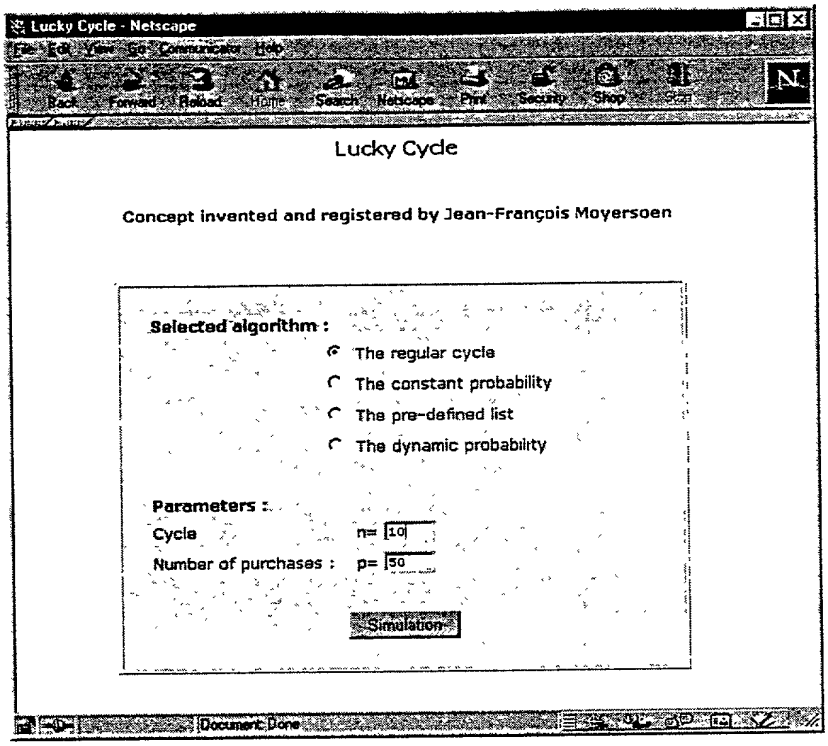
Lucky Cycle - Netscape
File Edit View Go Communicator Help
Back Forward Reload Home Search Netscape Print Security Shop Stop
Lucky Cycle
Concept invented and registered by Jean-François Moyersoen
Selected algorithm :
The regular cycle
The constant probability
The pre-defined list
The dynamic probability
Parameters :
Cycle n= 10
Number of purchases : p= 50
Simulation
Document Done

Fig. 6

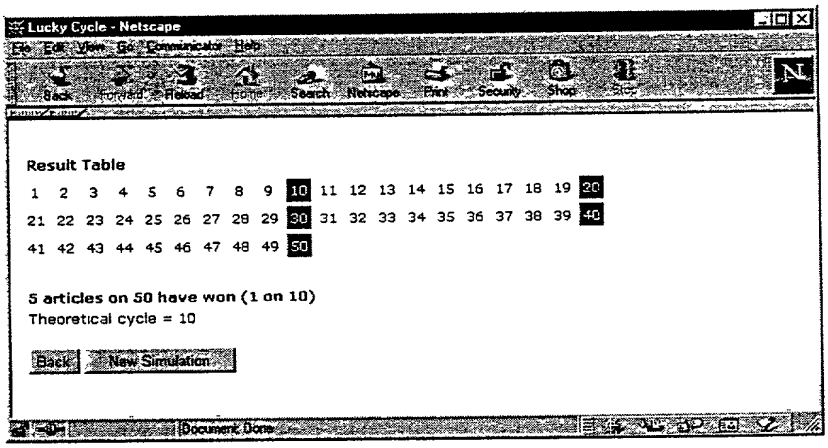
Lucky Cycle - Netscape
File Edit View Go Communicator Help
Back Forward Reload Home Search Netscape Print Security Shop Stop
Result Table
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40
41 42 43 44 45 46 47 48 49 50
5 articles on 50 have won (1 on 10)
Theoretical cycle = 10
Back New Simulation
Document Done

Fig. 7

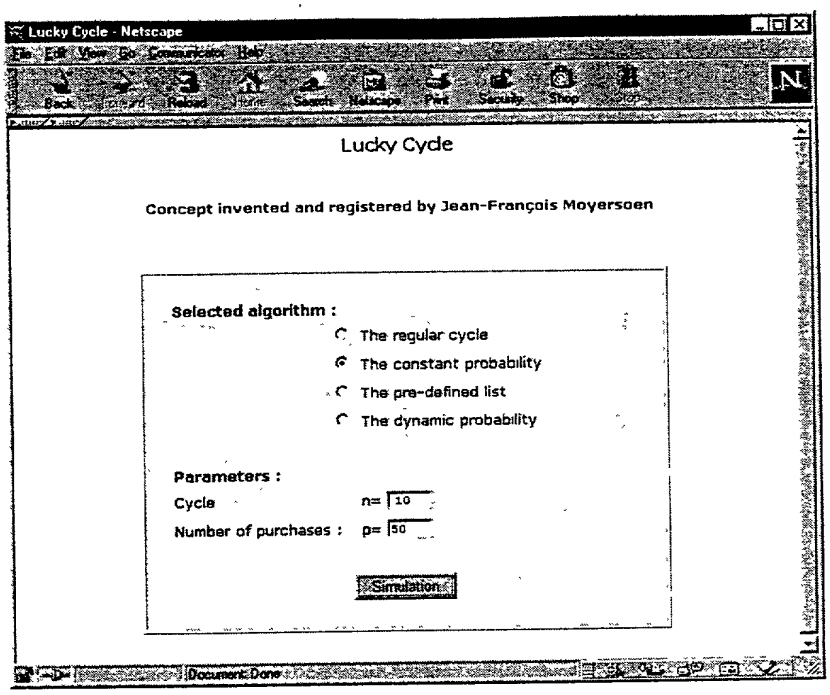
Lucky Cycle - Netscape
File Edit View Go Communicator Help
Back Forward Reload Home Search Netscape Print Security Shop Stop
Lucky Cycle
Concept invented and registered by Jean-François Moyersoen
Selected algorithm :
The regular cycle
The constant probability
The pre-defined list
The dynamic probability
Parameters :
Cycle n= 10
Number of purchases : p= 50
Simulation
Document: Done

Fig. 8

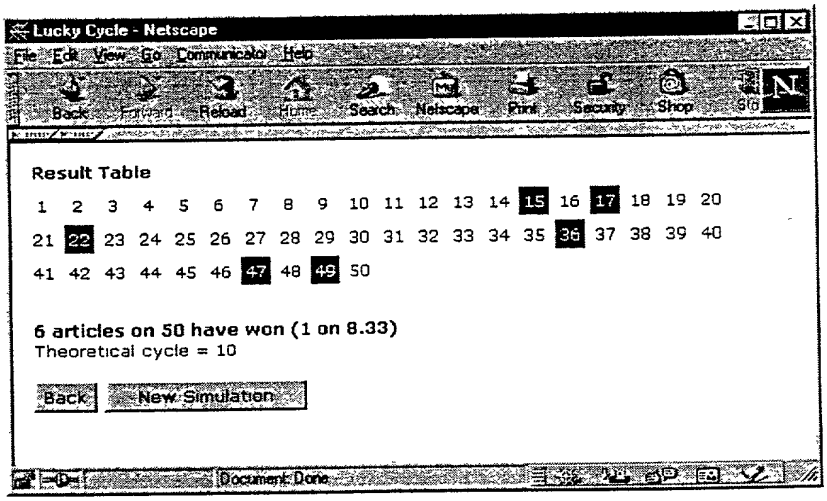
Lucky Cycle - Netscape
File Edit View Go Communicator Help
Back Forward Reload Home Search Netscape Print Security Shop
Result Table
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40
41 42 43 44 45 46 47 48 49 50
6 articles on 50 have won (1 on 8.33)
Theoretical cycle = 10
Back
New Simulation
Document: Done

Fig. 9

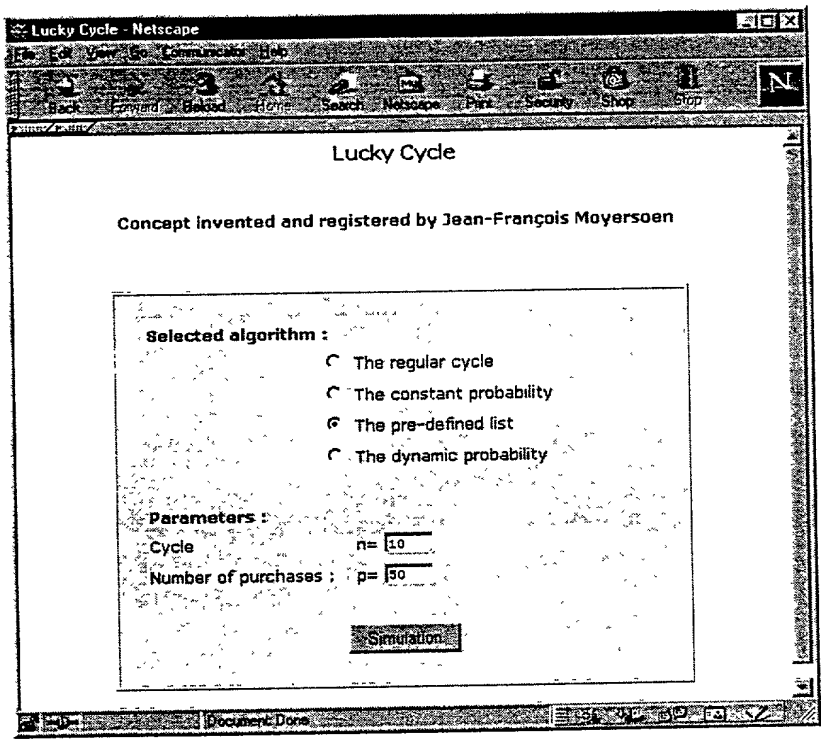
Lucky Cycle - Netscape
Lucky Cycle
Concept invented and registered by Jean-François Moyersoen
Selected algorithm :
The regular cycle
The constant probability
The pre-defined list
The dynamic probability
Parameters :
Cycle
n= 10
Number of purchases :
p= 50
Simulation
Document: Done

Fig. 10

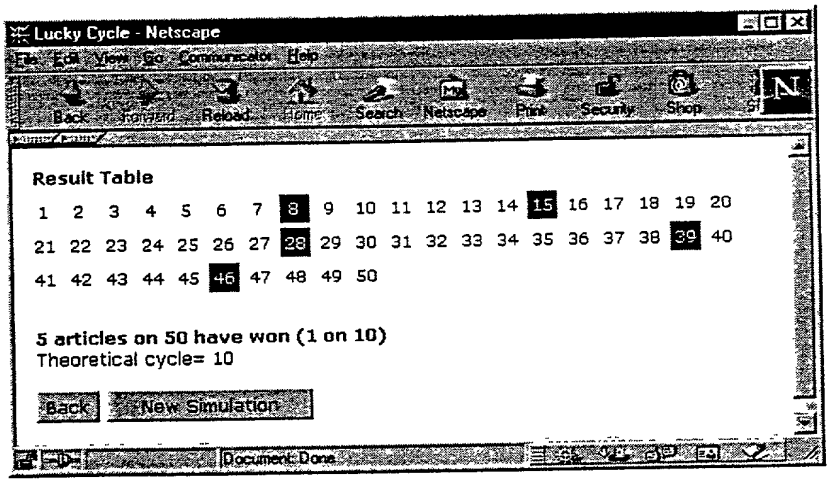
Lucky Cycle - Netscape
Result Table
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40
41 42 43 44 45 46 47 48 49 50
5 articles on 50 have won (1 on 10)
Theoretical cycle= 10
Back
New Simulation
Document: Done

Fig. 11

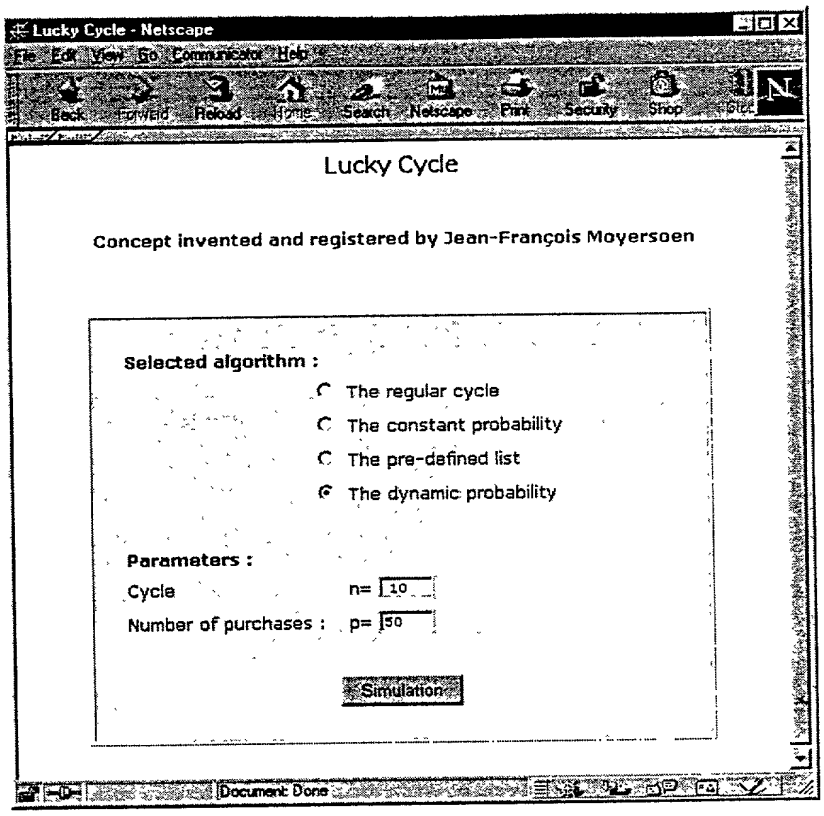
Lucky Cycle - Netscape
File Edit View Go Communicator Help
Back Forward Reload Home Search Netscape Print Security Shop Stop
Lucky Cycle
Concept invented and registered by Jean-François Moyersoen
Selected algorithm :
The regular cycle
The constant probability
The pre-defined list
The dynamic probability
Parameters :
Cycle n= 10
Number of purchases : p= 50
Simulation
Document Done

Fig. 12

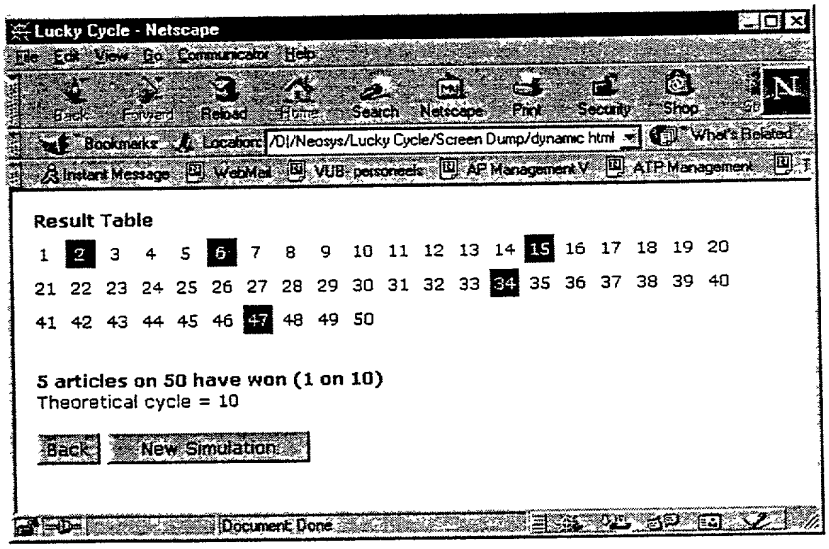
Lucky Cycle - Netscape
File Edit View Go Communicator Help
Back Forward Reload Home Search Netscape Print Security Shop
Bookmarks Location: /DI/Neosys/Lucky Cycle/Screen Dump/dynamic html What's Related
Instant Message WebMail VUB: personeels AP Management V ATP Management
Result Table
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40
41 42 43 44 45 46 47 48 49 50
5 articles on 50 have won (1 on 10)
Theoretical cycle = 10
Back New Simulation
Document Done

Fig. 13

UNITED STATES

PATENT APPLICATION DECLARATION AND POWER OF ATTORNEY – ORIGINAL APPLICATION	ATTORNEY'S DOCKET NO. 204,797

As a below named inventor, I hereby declare that:
My residence, post office address and citizenship are as stated below next to my name:
I verily believe I am the original, first and sole inventor (if only one name is listed below) or a joint inventor (if plural inventors are named below) of the invention entitled

(1) TITLE OF INVENTION: (1) METHOD OF OFFERING FREE PRODUCTS OR SERVICES OVER THE INTERNET

the specification of which

(2) CHECK APPROPRIATE BOX
(2) [X] is attached hereto.
[ ] was filed on _____ as Application No. _____
and was amended on _____(if applicable).

I hereby state that I have reviewed and understand the contents of the above-identified specification, including the claims, as amended by any amendment referred to above.
I acknowledge my duty to disclose information of which I am aware which is material to the patentability of this application under 37 CFR 1.56(a): the invention has not been pate issued before the date of this application in any country foreign tc by me or my legal representatives or assigns more than twelve mc for patents or inventor's certificate on the invention filed in any c application by me or my legal representatives or assigns.

STATEMENT OF FILING BY EXPRESS MAIL (37 CFR § 1.10)
This correspondence is being deposited with the United States Postal Service on September 06, 2000 in an envelope as "Express Mail Post Office to Addressee" Mailing Label Number EJ620981207US addressed to Director of the U.S. Patent and Trademark Office, Washington, D.C. 20231.

(3) CHECK APPROPRIATE BOX
(3) [ ] no such applications have been filed, or
[ ] such application(s) have been filed as follows:

EARLIEST FOREIGN APPLICATION(S), IF ANY, FILED WI				
Country	Application Number	Date (day, month, year)	(day, month, year)	Under 35 USC 119
U.S.A.	60/204,801	MAY 16, 2000		☒ Yes ☐ No
				☐ Yes ☐ No
				☐ Yes ☐ No
ALL FOREIGN APPLICATIONS, IF ANY, FILED MORE THAN 12 MONTHS PRIOR TO THIS APPLICATION				

I hereby claim the benefit under Title 35, United States Code § 120 of any United States application(s) listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States application in the manner provided by the first paragraph of Title 35, United States Code, § 112. I acknowledge the duty to disclose material information as defined in Title 37, Code of Federal Regulations, § 1.56(a) which occurred between the filing date of the prior application and the national or PCT international filing date of this application.

(5) COMPLETE DATA INDICATED IF APPLICABLE
(5) _____ (Application Serial No.) (Filing date) (Status: patented, pending, abandoned)
(5) _____ (Application Serial No.) (Filing date) (Status: patented, pending, abandoned)

Power of Attorney: As a named inventor, I hereby appoint the following attorney(s) and/or agent(s) to prosecute this application and transact all business in the Patent and Trademark Office connected therewith.

Jeffrey A. Schwab, Registration Number 24,490
Thomas E. Spath, Registration Number 25,928
Jay S. Cinamon, Registration Number 24,156
Michael I. Markowitz, Registration Number 30,659
Howard R. Jaeger, Registration Number 31,376
Anthony Coppola, Registration Number 41,493

Send correspondence to:
ABELMAN, FRAYNE & SCHWAB
150 East 42nd Street
New York, New York 10017-5612

Direct telephone calls to:
Jeffrey A. Schwab, Thomas E. Spath,
Jay S. Cinamon, Michael I. Markowitz,
Howard R. Jaeger or Anthony Coppola at (212) 949-9022

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment or both under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

(6) DETAILS REQUIRED FOR EACH INVENTOR

Full Name of Sole or First Inventor Jean-Francois MOYERSOEN	Inventor's Signature	Date
Residence 31, Avenue Princesse Grace, MC-98000 Monaco		Citizenship Monaco
Post Office Address Same as above		
Full Name of Second Joint Inventor, If Any	Inventor's Signature	Date
Residence		Citizenship
Post Office Address		
Full Name of Third Inventor, If Any	Inventor's Signature	Date
Residence		Citizenship
Post Office Address		
Full Name of Fourth Joint Inventor, If Any	Inventor's Signature	Date
Residence		Citizenship
Post Office Address		
Full Name of Fifth Joint Inventor, If Any	Inventor's Signature	Date
Residence		Citizenship
Post Office Address		
Full Name of Sixth Joint Inventor, If Any	Inventor's Signature	Date
Residence		Citizenship
Post Office Address		
Full Name of Seventh Joint Inventor, If Any	Inventor's Signature	Date
Residence		Citizenship
Post Office Address		